IN THE CLAIMS

1. (Currently Amended) A method, comprising:

programming, based on predefined data, one or more fields of configuration registers of a peripheral component interconnect (PCI) bridge device having a first identification (ID) ~~in response to a configuration cycle of~~during an initialization period of a data processing system, the programmed PCI bridge having a second ID different than the first ID, the one or more fields of the configuration registers including at least one identification register for identifying the ~~peripheral~~ PCI bridge device; ~~and~~

reporting the PCI bridge device identified by the second ID to the data processing system during a configuration cycle of the data processing system; and

operating the PCI bridge ~~peripheral~~ device using at least one programmed identification register representing the second ID.

2. (Currently Amended) The method of claim 1, further comprising enabling the PCI bridge ~~peripheral~~ device in the data processing system after the programming, the PCI bridge ~~peripheral~~ device having ~~an identification~~the second ID derived from the at least one identification register programmed with the predefined data.

3. (Currently Amended) The method of claim 2, further comprising verifying content of the one or more programmed registers against the predefined data before enabling the PCI bridge ~~peripheral~~ device.

4. (Currently Amended) The method of claim 2, wherein the PCI bridge ~~peripheral~~ device is ~~a PCI (peripheral component interconnect) compatible device and the enabled peripheral device operates in compliance with a PCI specification~~implemented as a part of a chipset of the data processing system.

5. (Currently Amended) The method of claim 1, further comprising retrieving the predefined data from a memory associated with the PCI bridge ~~peripheral~~ device during the initialization period of the data processing system prior to programming the peripheral device, content of the programmed at least one identification register being persistent until a next initialization period of the data processing system.

6. (Currently Amended) The method of claim 1, further comprising:

determining whether identification of the ~~peripheral~~ PCI bridge device needs to be programmed using the predefined data;

programming the at least one identification registers of the PCI bridge ~~peripheral~~ device using the predefined data, if the identification of the peripheral device needs to be programmed using the predefined data; and

loading the at least one identification registers using default data if the peripheral device's identification does not need to be programmed.

7. (Currently Amended) The method of claim 6, wherein whether the identification of the PCI bridge ~~peripheral~~ device needs to be programmed is determined based on one or more bit patterns of the predefined data.

8. (Currently Amended) The method of claim 1, wherein the at least one identification register includes at least one of vendor ID, device ID, and revision registers, and wherein the initialization period comprises a PRESET period of the data processing system.

9. (Currently Amended) The method of claim 1, further comprising configuring the at least one identification register as a read only register after the PCI bridge ~~peripheral~~ device is enabled in response to the configuration cycle of the data processing system.

10. (Original) The method of claim 1, further comprising detecting a reset signal received at the peripheral device, programming the one or more fields of configuration registers of the peripheral device using the predefined data being automatically performed in response to the reset signal.

11. (Currently Amended) A machine-readable medium having executable code to cause a machine to perform a method ~~for power management~~, the method comprising:

programming, based on predefined data, one or more fields of configuration registers of a peripheral component interconnect (PCI) bridge device having a first identification (ID) ~~in response to a configuration cycle~~during an initialization period of a data processing system, the programmed PCI bridge having a second ID different than the first ID, the one or more fields of the configuration registers including at least one identification register for identifying the PCI bridge ~~peripheral~~ device; ~~and~~

reporting the PCI bridge device identified by the second ID to the data processing system during a configuration cycle of the data processing system; and

operating the PCI bridge ~~peripheral~~ device using at least one programmed identification register representing the second ID.

12. (Currently Amended) The machine-readable medium of claim 11, wherein the method further comprises enabling the PCI bridge ~~peripheral~~ device in the data processing system

after the programming, the PCI bridge ~~peripheral~~ device having ~~an identification~~the second ID derived from the at least one identification register programmed with the predefined data.

13. (Currently Amended) The machine-readable medium of claim 12, wherein the method further comprises verifying content of the one or more programmed registers against the predefined data before enabling the PCI bridge ~~peripheral~~ device.

14. (Currently Amended) The machine-readable medium of claim 12, wherein the PCI bridge ~~peripheral~~ device is implemented as a part of a chipset of the data processing system~~a PCI (peripheral component interconnect) compatible device and the enabled peripheral device operates in compliance with a PCI specification~~.

15. (Currently Amended) The machine-readable medium of claim 11, wherein the method further comprises retrieving the predefined data from a memory associated with the PCI bridge ~~peripheral~~ device during the initialization period of the data processing system prior to programming the peripheral device, content of the programmed at least one identification register being persistent until a next initialization period of the data processing system.

16. (Currently Amended) The machine-readable medium of claim 11, wherein the method further comprises:

determining whether identification of the PCI bridge ~~peripheral~~ device needs to be programmed using the predefined data;

programming the at least one identification registers of the PCI bridge ~~peripheral~~ device using the predefined data, if the identification of the peripheral device needs to be programmed using the predefined data; and

loading the at least one identification registers using default data if the peripheral device's identification does not need to be programmed.

17. (Currently Amended) The machine-readable medium of claim 16, wherein whether the identification of the PCI bridge ~~peripheral~~ device needs to be programmed is determined based on one or more bit patterns of the predefined data.

18. (Currently Amended) The machine-readable medium of claim 11, wherein the at least one identification register includes at least one of vendor ID, device ID, and revision registers, and wherein the initialization period comprises a PRESET period of the data processing system.

19. (Currently Amended) The machine-readable medium of claim 11, wherein the method further comprises configuring the at least one identification register as a read only register after the PCI bridge ~~peripheral~~ device is enabled in response to the configuration cycle of the data processing system.

20. (Original) The machine-readable medium of claim 11, wherein the method further comprises detecting a reset signal received at the peripheral device, programming the one or more fields of configuration registers of the peripheral device using the predefined data being automatically performed in response to the reset signal.

21. (Currently Amended) A ~~peripheral~~ PCI bridge device, comprising:

a processor to perform one or more peripheral functions;

one or more programmable configuration registers accessible by the processor, the one or more programmable configuration registers including at least one identification register for identifying the peripheral device; and

a memory coupled to the processor to store predefined data, the predefined data being used to program the one or more programmable configuration registers

including the at least one identification register during an initialization period of the processor to change an identification (ID) of the PCI bridge device from a first ID to a second ID different than the first ID, wherein the PCI bridge device is represented using the second ID in response to a configuration cycle of the ~~peripheral~~ PCI bridge device, such that the PCI bridge device is configured to operate using the second ID during a normal cycle of the PCI bridge device.

22. (Original) The peripheral device of claim 21, wherein the at least one identification register includes at least one of vendor ID, device ID, and revision registers.

23. (Original) The peripheral device of claim 21, wherein the memory is a serial read-only memory (SROM).

24. (Currently Amended) A data processing system, comprising:

one or more processors;

a bus coupled to the one or more processors;

~~one or more peripheral devices~~a PCI bridge device coupled to the bus for interfacing a PCI bus with the bus, ~~at least one of the peripheral devices~~the PCI bridge device including

one or more functional units to perform one or more peripheral functions,

one or more programmable configuration registers including at least one identification register for identifying the respective peripheral device, and

a memory to store predefined data, the predefined data being used to program the one or more programmable configuration registers including the at least one identification register during an initialization period of the

processor to change an identification (ID) of the PCI bridge device from a first ID to a second ID different than the first ID, wherein the PCI bridge device is represented using the second ID in response to a configuration cycle of the peripheral device, such that the PCI bridge device is configured to operate using the second ID during a normal cycle of the PCI bridge device.

25. (Original) The data processing system of claim 24, wherein the at least one identification register includes at least one of vendor ID, device ID, and revision registers.

26. (Original) The data processing system of claim 24, wherein the memory is a serial read-only memory (SROM).

27. (Currently Amended) The data processing system of claim 24, wherein the ~~one or more peripheral devices are PCI (peripheral component interconnect) compatible devices~~PCI bridge device is implemented as a part of a chipset of the data processing system.